

## # ADORDD

\*\*\*\*\*

ADORDD for (x)Harbour its ready!

PLEASE MAKE A DONATION!

We will all profit for sure from such development so I think it's fair to ask everyone to contribute with a minimal importance of 20 Euros (because of the PayPal costs) for a good cause.

Go to <http://ajusera.com/> scroll down and click PayPal button "Doar"

Or

You have bank details TRANSFERÊNCIA BANCÁRIA = BANK TRANSFER

You can transfer to those account details

They can send you a contribution receipt if you send them a email with your details.

This is a non-profit organization that performs an important social work supporting the elderly, young problematic, food distribution to the needy, etc. This organization is based on the work of Father Jeronimo Usera (Spanish) well known in Spain, Portugal and South America.

You can get more information at [ajusera.com](http://ajusera.com) (sorry its only in portuguese)

Steps to start working with **ADORDD**:

=====

**Step 1:** Just add adordd.prg to your project and include adordd.ch

**Step 2:** Set adordd parameters (see in adordd.ch for syntax) and add it in your main.prg like this:

```
// Load ADORDD
REQUEST ADORDD, ADOVERSION, RECSIZE
RddRegister( "ADORDD",1 )
RddSetDefault( "ADORDD" )
```

## A) SET ADODBF TABLES INDEX LIST TO...

---

It must contain all your actual indexes and expressions per table.

Example:

```
SET ADODBF TABLES INDEX LIST TO {  
  {"TABLE1", {"FIRST", "FIRST"} }, {"TABLE2"  
  , {"CODID", "STR(CODID, 2, 0)"} } }
```

## **VERY IMPORTANT**

**ALL ARRAYS ELEMENTS MUST BE IN UPPERCASE  
OTHERWISE INDEX WILL NOT WORK**

## B) SET ADO INDEX UDFS TO...

Here you must place the UDFs index expressions.

Indexes with UDFs are expensive to create and maintain so you should only place here your UDFs and index expressions that change the field length or value. Data type conversion expressions (the most used) should not be placed here.

Example:

```
SET ADO INDEX UDFS TO {"&", "SUBSTR", "PAD", "MYFUNC"}
```

## C) SET ADO TEMPORAY NAMES INDEX LIST TO...

---

Indicates the names used for temporary files at SQL level.

It must start by TMP or TEMP but can be "TMPROGER"

These temporary files are mainly used for temporary indexes created in the SQL server as TEMPORARY and automatically destroyed after connection ends.

They are only visible to the user that created them.

Example:

```
SET ADO TEMPORAY NAMES INDEX LIST TO  
{ "TMP", "TEMP" }
```

#### **D) SET ADO FIELDRECNO TABLES LIST TO ...**

---

This Set lets you indicate a different autoinc field of the default per table to be used as recno().

Example:

```
SET ADO FIELDRECNO TABLES LIST TO  
{ { "TABLE1", "MYHBRECNO1" }, { "TABLE2", "MYHBRECNO2" } }
```

#### **E) SET ADO DEFAULT RECNO FIELD TO...**

---

This Set indicates the default field name to be used as recno in all tables besides the mentioned above.

Example:

```
SET ADO DEFAULT RECNO FIELD TO "HBRECNO"
```

#### **ATTENTION:**

**The D and or E sets are absolutely necessary and without them you will get an error trying to opening the table.**

#### **F) SET ADO FIELDDELETED TABLES LIST TO...**

This Set lets you indicate a different logical field of the default per table to be used as deleted flag.

Example:

```
SET ADO FIELDDELETED TABLES LIST TO  
{ {"TABLE1", "HBMUDELETE1"}, {"TABLE2", "HBMUDELET  
E2"} }
```

### **G) SET ADO DEFAULT DELETED FIELD TO...**

This Set indicates the default field name to be used as deleted flag in all tables besides the mentioned above.

Example:

```
SET ADO DEFAULT DELETED FIELD TO "HBDELETED"
```

### **ATTENTION:**

**The F and or G sets are absolutely necessary and without them you will get an error trying to opening the table.**

### **H) SET ADO DEFAULT DATABASE TO ... SERVER TO ... ENGINE TO ... USER TO ... PASSWORD TO ... CLASSNAME ...**

-----

This Set indicates the default server and database and authentication parameters we are using.

Connection gets established here.

The engines supported by ADORDD are ACCESS, MYSQL, ORACLE, INFORMIX, MSSQL, FIREBIRD, POSTGRE, ANYWHERE, DBASE, SQLITE, FOXPRO , ADS

Example:

```
// Access  
SET ADO DEFAULT DATABASE TO  
"D:\LUCAS\TEST2.mdb"  
SERVER TO "" ENGINE TO ACCESS USER TO ""  
PASSWORD TO ""
```

```
// MySQL
SET ADO DEFAULT DATABASE TO cDataBase SERVER
TO cServer ENGINE TO MYSQL USER TO cUser
PASSWORD TO cPassWord
```

**The CLASSNAME it's for future use. It allows having another class providing all ADO methods and Data to access through it the SQL engine.**

### **G) SET ADO LOCK CONTROL SHAREPATH TO ... RDD TO ...**

-----

This set enables ADORDD to assure locking records and exclusive use of files as any other RDD.

**If you want to leave the SQL engine take care of all locking you can use OFF but be aware that this does not conform with ISAM programing:**

```
SET ADO FORCE LOCK OFF
```

You need to supply a path where ADORDD creates the tlocks.dbf file to control this.

This RDD file must be a RDD working with locks such as DBFCDX.

This is not a SQL table and if you need to work in WAN and need lock control you will need to:

- a) The connection to SQL server
- b) Ex a VPN where you can access this share.

Example:

```
SET ADO LOCK CONTROL SHAREPATH TO "D:\LUCAS"
RDD TO "DBFCDX"
```

**Step 3:** You can start working with ADORDD like a DBFCDX

**and thats it!**

**Now if you need to upload your tables to any SQL you can do it with:**

```
use "table" via "dbfcdx"

copy "table" to "sqtable" via "adordd"

use "sqltable"

//you can use a table in a new connection

Use "ctable@connection string" alias
"whatever"
```

**This is a low speed method and can take long time if the tables are huge. In this case you have to find yourself another way to upload data. You can use for example adofuncs.prg to import dbfs.**

**If you need to add deleted and/or recno field to your existing SQL tables you can do it like this:**

```
oCon := hb_GetAdoConnection()

aTables := hb_adoRddGetTables( oCon )

FOR n := 1 to LEN( aTables )

    TRY // if already exist ignore it and continue

        oCon:Execute("ALTER TABLE "+( aTables[ n ] )+"
                        ADD COLUMN HBDELETE BIT DEFAULT 0")

        oCon:Execute("ALTER TABLE "+( aTables[ n ] )+"
                        ADD COLUMN HBRECNO AUTOINC PRIMARY
                        KEY")

    CATCH

END

NEXT
```

### **Functions you can call in adordd.prg to use in our application :**

ADOVERSION() Returns adordd version

ADOBEGINTRANS(nWa)

ADOCOMMITTRANS(nWa)

ADOROLLBACKTRANS(nWa)

hb\_adoRddGetConnection( nWorkArea ) Returns the connection for the workarea

hb\_adoRddGetRecordSet( nWorkArea ) Returns the recordset for the nWorkArea

hb\_adoRddGetTableName( nWorkArea ) Returns table name for the nWorkArea

hb\_adoRddExistsTable( oCon, cTable, cIndex, cView ) Returns .t. if exist on the DB

hb\_adoRddDrop( oCon, cTable, cIndex, cView, DBEngine ) Drops (delete) table index or view in the DB

hb\_GetAdoConnection() Returns ado default connection

hb\_adoRddGetTables( oCon ) Returns all tables in the database in an array